

《C/C++ 学习 指南》

第24.1讲：拷贝构造函数

作者：邵发 QQ群：417024631

官网：<http://www.afanihao.cn/>

习题：<http://www.afanihao.cn/kbase/>

本书配套纸质教材，具体信息见官网。

C/C++学习指南 邵发 www.afanihao.cn

背景知识

本章将使用“引用”的语法，请学习/回顾第14章。

拷贝构造函数

拷贝构造函数是一种特殊的构造函数。copy constructor

- (1) 它是构造函数，所以函数名是类名、没有返回值
- (2) 它是特殊的构造函数：参数形式是固定的

```
class Object
{
public:
    Object( const Object& other );
};
```

拷贝构造函数的意义

拷贝构造函数的含义： 以一个对象为蓝本，来构造另一个对象。

```
Object b;
Object a(b); // 或写成 Object a = b;
```

称作：以b为蓝本，创建一个新的对象a。

(a是b的一个拷贝/备份，两者内容完全相同)

拷贝构造函数的调用

拷贝构造函数从来不显式调用，而是由编译器隐式地调用。
在以下三种情况：

(1) 定义对象

```
Object a;  
Object b(a); // 或写成 Object b = a;
```

(2) 动态创建对象

```
Object a;  
Object* p = new Object(a);
```

(3) 函数的传值调用

```
void Test(Object obj);
```

注意 (1): 区分“构造”与“赋值”

构造：

```
Object a;  
Object b = a; // 或写作 Object b(a);  
// 此为“构造”，在创建对象的时候给初值，拷贝构造函数被调用
```

赋值：

```
Object a(1, 2);  
Object b;  
b = a; // 此为“赋值”，不会调用拷贝构造函数
```

注意 (2): 可以访问同类对象的private成员

在拷贝构造函数，可以访问参数对象的任意成员

因为它们是同类，所以访问不受限制。

```
Object(const Object& other)
{
    this->a = other.a;
    this->b = other.b;
}
```

默认的“拷贝构造函数”

当没有书写拷贝构造函数时，编译器会默认提供一个拷贝构造函数。默认的拷贝动作：**将每一个成员逐个拷贝。**

也就是说，在多数情况下，我们无需操心，编译器会帮我们完成成员的拷贝动作。

(VC演示)

问题：**既然默认拷贝成员是够用的，为什么还要自己写一个拷贝构造函数呢？**

拷贝构造函数的检查项

注意：除非必要，不要添加拷贝构造函数。

一旦你决定了要添加拷贝构造函数，请仔细检查：

(1) 所有的成员变量，要依次拷贝
所有成员变量，不要遗漏

(2) 调用父类的拷贝构造函数

(要么不负责，要么负责全部事情)

问题：拷贝构造函数非常麻烦，容易遗漏，为什么还要自己写一个拷贝构造函数呢？

小结

1. 拷贝构造函数的语法：（特点）
2. 拷贝构造函数在何时会被调用？
3. 编译器已经默认提供了一个好用的拷贝构造函数
4. 拷贝构造函数的检查项：务必仔细检查，不要遗漏
5. 除非必要，不要添加拷贝构造函数。实际上，只在一种情况下，才需要添加拷贝构造函数（下一讲）